
Documentation: Windows Documentation

Release 0.1.0

Joon Ro

Nov 20, 2017

Contents

1	Move C:\Users directory to D:\Users	3
1.1	Dropping to the Command Line Prompt	3
1.2	Figure out drive letters	4
1.3	Copy Users folder to D:\Users	4
1.4	Create the symbolic link	4
1.5	Using BitLocker	5
2	Admin	7
2.1	Environmental Variables	7
2.2	Symbolic Links	7
2.3	PowerShell Remoting	8
2.4	SSH	8
2.5	Tinkerer blog	9
2.6	9
3	Installation	11
3.1	SysAdmin	11
3.2	Browsers	11
3.3	Cloud	12
3.4	Devel	12
3.5	Office	12
3.6	Multimedia	12
3.7	Communications	12
3.8	Utils	13
3.9	Chocolatey	13
3.10	Libraries	13
3.11	Utilities	13
3.12	Multimedia	14
3.13	Preview Handlers	14
3.14	Fonts	15
4	PowerShell	17
4.1	Determine what version of PowerShell is installed	17
4.2	Install Modules	17
4.3	Persistent History	17
4.4	PSReadline	18
4.5	Text	18

4.6	Control Processes	18
4.7	GNU/Linux commands equivalents	18
4.8	PowerShell Community Extensions (PSCX)	19
4.9	Remoting	19
5	Python Environment	25
5.1	Resources	25
6	Anaconda Installation	27
6.1	Download	27
6.2	Mayavi	27
6.3	ViTables	28
6.4	Update Packages	28
7	Devel Environment	31
7.1	General	31
7.2	Git	31
7.3	PowerShell	31
7.4	Terminal	32
7.5	Libraries	32
7.6	Compilers	32
8	LaTeX/LyX	35
8.1	Global location of .bib file.	35
8.2	Inverse search with LyX	35
8.3	Forward search with LyX	37
8.4	Mendeley Settings	37
9	Backup	39
9.1	LyX settings	39
9.2	Outlook	39
9.3	Windows 8 Apps	39
10	Troubleshooting	41
10.1	General	41
10.2	Skydrive	41
10.3	Network	42
10.4	0xd0000225 switch to live account	42
11	Indices and tables	43

Contents:

Move C:\Users directory to D:\Users

Warning: This is not recommended anymore: <http://support.microsoft.com/kb/949977/en-us>

When I use Windows, I always have separate partition for the data (D:). I have never tried to actually move the whole Users folder to D: though. Instead I moved folders with data such as Documents to D: manually after installing Windows.

After purchasing a couple of SSD's for my laptops, I decided to install Windows 8 on them, and decided to use more sophisticated (and elegant) method of moving Users folder to D:, which leverages the soft link functionality of Windows.

I have read some howto's about this and at first they look complicated, but it turns out it is pretty simple (about 3 lines of commands) and robust. As usual, it was so much better than manual way.

There are just a couple of things which can mess things up if you are not careful, and I wanted to share the experience so others can do this without trial and error.

I assume you already have separate partition D:.

1.1 Dropping to the Command Line Prompt

1.1.1 Windows 8

Since no process should be using files in the C:\Users directory, you have to drop to the recovery mode command line prompt. There are two ways that I know of. You can do it by booting with installation DVD and also you can do it in installed Windows 8. I recommend the second method as it is much faster. It became very easy in Windows 8 since you can invoke this within running Windows.

Please see the following link for how-to: [How to start the Recovery Environment Command Prompt in Windows 8](#)

1.1.2 Windows 7

Please see the following website for how-to: [How to Get to the Recovery Console in Windows](#)

1.2 Figure out drive letters

This is the first (a little bit) tricky part. For some reason when you drop to this command prompt the drive letters are different. In my case, it was always X: which was the recovery partition, and C: became D: and D: became E:. So when you are working in D: in the prompt, actually you are at C:. This can be confusing but once you figure it out it is not that bad. Anyway, if your drive letter mapping is different from mine, you have to adjust the following commands accordingly. Basically, when you make copies you use the letters assigned in the command prompt, but when you make the link, you have to use the letters that windows uses.

1.3 Copy Users folder to D:\Users

Two things are important in this step: 1) you have to use the correct drive letters, and 2) you have to use the exact options for `robocopy`. If you mess up 1), you won't be able to log in, and if you mess up 2), then you might be able to log in but will have other problems.

Do the following to copy C:\Users to D:\Users. Again, keep in mind the drive letters are remapped:

```
X:> D:
D:> robocopy /copyall /mir /xj Users E:\Users
```

`/mir` option will delete all files in the target if they do not exist in the source. Hence, if you already have D:\Users and want to keep the files, then use `/E` option:

```
D:> robocopy /copyall /E /xj Users E:\Users
```

Again, here D: is C: in actual, and E: is D: in actual. Among `robocopy` options, if you miss `/copyall`, then it will not copy the permission information, and after log-in your Windows 8 apps will crash upon launching. Make sure you get 0 for failed column in the `robocopy` results output.

After verifying there were no error, do the following to remove the original C:\Users directory:

```
D:> rmdir /S /Q Users
```

1.4 Create the symbolic link

This is the final step, creating the symbolic link at C:\Users which points to D:\Users. Here the drive letter thing again comes up. Since the symbolic link will be used in actual windows session, you have to use the actual drive letter, "D:" as the target. So you end up making a symbolic link at D:\Users (command prompt letter - C: in actual) whose target is also D:\Users. Since you are already at D:, use the following so it is less confusing:

```
D:> mklink /J Users D:\Users
Junction created for Users <<====>> D:\Users
```

I have not tried other options, but it seems if you don't use `/J` (Junction) option you will have to do a registry hack.

You can verify the link with `dir` command. You will see:


```
<JUNCTION>      Users [D:\Users]
```

Anyway, that's it. When you exit and reboot (select continue to windows 8), you will be able to log in, and when you open up the explorer, you will see that `C:\Users` is a symbolic link which points to `D:\Users.`:

```
D:> exit
```

Again, the steps are very easy when you figure it out (especially the drive letter thing) but at the same time it is very easy to make a mistake.

1.5 Using BitLocker

You can also use BitLocker.

1. Encrypt `C:` with BitLocker
2. Encrypt `D:` with BitLocker, and make sure to enable Auto-Unlock

Then when boot, Windows will unlock `D:` so you can log in.

2.1 Environmental Variables

- System environmental variable:

```
setx -m PATH "%PATH%;.\"  
setx -m PATH "%PATH%;C:\Program Files (x86)\KDE\bin"
```

- Local environmental variables:

```
setx HOME "C:\Users\Joon"  
setx PATH "%PATH%;%HOME%\bin"
```

2.2 Symbolic Links

```
> mklink /J c:\home %home%..\..  
> mklink /J %home%\lyx %appdata%\LyX2.0  
> mklink /J %home%\mozilla\firefox "%appdata%\Mozilla\Firefox"  
> mklink /J %home%\config\sublime-text-3 "%appdata%\Sublime Text 3"  
> mklink /J %home%\vimfiles "%home%\vim"  
> mklink /J %home%\kde4 "%appdata%\kde"  
> mklink /J %home%\wingide5 "%appdata%\Wing IDE 5"  
> mklink /J C:\lib D:\lib  
> mklink /J C:\lib64 D:\lib64  
  
> mklink /J Documents D:\Users\joon\Documents  
> mklink /J Downloads D:\Users\joon\Downloads  
> mklink /J Music D:\Users\joon\Music  
> mklink /J Pictures D:\Users\joon\Pictures  
> mklink /J Videos D:\Users\joon\Videos  
  
> mklink /J Copy D:\Users\joon\Copy
```

```
> mklink /J Comics D:\Users\jjoon\Comics
> mklink /J Data D:\Users\jjoon\Data
> mklink /J AeroFS D:\Users\jjoon\AeroFS
> mklink /J Dropbox D:\Users\jjoon\Dropbox
> mklink /J ".\Google Drive" "D:\Users\jjoon\Google Drive"
```

2.3 PowerShell Remoting

To configure the computer (the server) to receive Windows PowerShell remote commands, run the following in an elevated PowerShell:

```
Enable-PSRemoting
```

2.4 SSH

2.4.1 SSH Server

- Download and install [Bitwise SSH Server](#):

```
choco install winsshd
```

2.4.2 SSH Client

- Download [Bitwise SSH Client \(Tunnelier\)](#):

```
choco install Tunnelier
```

- You can use `ssh` comes with Git Bash as well.

2.4.3 Bitwise SSH Server Public key authentication

- Generate public/private rsa key pair:

```
ssh-keygen
```

- To be able to add public keys to the server from the client, download [Bitwise SSH Server Remote Control Panel](#).
- Then follow instructions here: <http://www.bitwise.com/wug-publickey>

1. Login to the server
2. Open Bitwise SSH Server Control Panel
3. Open Easy Settings
4. 2. Windows accounts
5. From the Public keys imported tab, you can import local public keys. (Import both bitwise-generated and ssh-keygen generated keys)

2.5 Tinkerer blog

```
$ cd ~/git
$ git clone git@github.com:joonro/blog.git
$ mklink /J en C:\Users\joon\Dropbox\tinkerer\en\blog\html
$ mklink /J ko C:\Users\joon\Dropbox\tinkerer\ko\blog\html
```

2.6

The location of in Windows is:

```
%APPDATA%\..\LocalLow\NPKI
```


3.1 SysAdmin

- Chocolatey:

```
@powershell -NoProfile -ExecutionPolicy unrestricted -Command "iex ((new-object net.webclient).DownloadString('https://chocolatey.org/install.ps1'))" && SET-  
-PATH=%PATH%;%systemdrive%\chocolatey\bin
```

```
choco install Tunnelier
```

- Windows Cron Service
- Paragon ExtFS for Windows : ext4 Read/Write support
- CrashPlan
 - See How to Move Your CrashPlan Backups to a New Computer to restore your backup settings
- Tiny Windows Borders

```
choco install LinkShellExtension f.lux pscx wifi-manager
```

- BgInfo

3.2 Browsers

```
choco install Firefox GoogleChrome # browsers
```

3.3 Cloud

```
choco install dropbox googledrive
```

3.4 Devel

```
choco install Emacs geany nano KickAssVim SublimeText3 # Editors; KickAssVim will  
↳pull ctags and vim  
choco install consolez git.install meld winmerge # devel  
choco install SourceCodePro # fonts; this doesn't work it seems
```

3.4.1 Pinning Emacs to the taskbar

<http://superuser.com/questions/259146/why-latest-emacs-version-dont-support-windows-7-taskbar>

If you just pin Emacs icon on the taskbar, next time you click it to launch Emacs, it will have a terminal window as well as GUI window. In order to fix this,

1. Run `runemacs.exe` with no pre-existing icon on the taskbar.
2. Right click on the running Emacs icon in the taskbar, and click on “pin this program to taskbar” item.
3. Close the Emacs
4. Shift right-click on the pinned Emacs icon on the taskbar, click on Properties, and change the target from `emacs.exe` to `runemacs.exe`.

3.5 Office

```
choco install PDFXchangeEditor PDFCreator # PDF
```

- It is better to install [Sumatra PDF](#) from the installer so you can install the browser plugins and pdf preview.

3.6 Multimedia

```
choco install foobar2000 mpc-hc
```

3.7 Communications

```
choco install winscp deluge youtube-dl
```


3.8 Utils

```
choco install 7zip autohotkey_1 dexpot teamviewer # utils
```

3.9 Chocolatey

```
choco install IcoFx
choco install SourcePreviewHandler
choco install keepass # pims
choco install paint.net gimp

choco install pandoc
```

3.10 Libraries

3.10.1 GTK

In both cases, get the all-in-one bundle.

- for Windows (32-bit) (Get the 2.x version)
- for Windows (64-bit)

Set PATH:

```
setx -m PATH "%PATH%;C:\lib\gtk\bin"
```

3.11 Utilities

3.11.1 unison

3.11.2 Aspell

1. Download Aspell executable (Full installer) and at least one dictionary (aspell-en-0.50-2-3.exe) from <http://aspell.net/win32/>.
2. Install Aspell-0-50-3-3-Setup and then Aspell-en-0.50-2-3.

3.11.3 awk

- Download and install Gawk for Windows
- Put C:\Program Files (x86)\GnuWin32\bin\ into your Path:

```
setx -m PATH "%PATH%;C:\Program Files (x86)\GnuWin32\bin"
```

3.11.4 cURL

- If you have [Chocolatey](#) installed (which I highly recommend):

```
$ choco install curl
```

- Otherwise, follow instructions [here](#).
- Also, follow the instructions [here](#) to add Certificate Authority Public Keys for https.

3.11.5 ClassicShell

- Get the best looking Start Menu for Classic Shell 4+ with [Winaero Skin 2.0](#)
- [windows 8.1 ms clone button](#)

3.11.6 Others

- [Clipboard Help+Spell](#)
- [Free Alarm Clock](#)
- [DiskInternals Linux Reader](#)
- [SQLiteSpy](#)

3.12 Multimedia

3.12.1 foobar2000

- [Columns UI](#)
- [Quick Tagger](#)
- [flac](#)
- [mplayer2](#)

3.13 Preview Handlers

3.13.1 PDF Preview

For PDF Preview in Explorer and Directory Opus, get [PDF XChange Viewer](#) and install Shell Extensions. It is by default checked when you install it. Portable version does not have this plugin.

See http://www.tracker-software.com/shell_ext.html for explanations.

3.13.2 Source Code

- [Source Preview Handler](#)

3.14 Fonts

-

- If scripts are not enabled, run PowerShell as Administrator and call:

```
Set-ExecutionPolicy RemoteSigned -Scope CurrentUser -Confirm
```

- Install PowerShell Community Extensions

4.1 Determine what version of PowerShell is installed

```
$psversiontable.psversion
```

4.2 Install Modules

4.2.1 PowerShell Resource Gallery

<https://msconfiggallery.cloudapp.net/>

```
Install-Module -Name PSReadline  
Install-Module -Name PSReadline -Scope CurrentUser
```

4.3 Persistent History

```
$MaximumHistoryCount = 31KB  
$PoshHistoryPath = "$home\_posh_history.xml"  
  
# Load history if history file exists  
if (Test-path $PoshHistoryPath)
```

```
{ Import-CliXml $PoshHistoryPath | Add-History }

# Save history on exit, remove duplicates
Register-EngineEvent PowerShell.Exiting {
    Get-History -Count $MaximumHistoryCount | Group CommandLine |
    Foreach {$_ .Group[0]} | Export-CliXml "$Home\_posh_history.xml"
} -SupportEvent

# hg function to search history
function hg($arg) {
    Get-History -c $MaximumHistoryCount | out-string -stream |
    select-string $arg
}
```

4.4 PSReadline

```
# PSReadline
Import-Module PSReadLine
Set-PSReadlineKeyHandler -Key UpArrow -Function HistorySearchBackward
Set-PSReadlineKeyHandler -Key DownArrow -Function HistorySearchForward
Set-PSReadlineKeyHandler -Key Tab -Function MenuComplete
```

4.5 Text

4.5.1 Write a file in UTF-8 without the BOM

<http://stackoverflow.com/questions/5596982/using-powershell-to-write-a-file-in-utf-8-without-the-bom>

Out-File -Encoding "UTF8" force the BOM when using UTF-8. Use the following:

```
[System.IO.File]::WriteAllLines($File, $Contents)
```

4.6 Control Processes

Get-Process shows current processes. You can kill a process with Stop-Process and process ID:

```
Get-Process ProcessName | Stop-Process
```

For example:

```
Get-Process ClipboardHelpAndSpell | Stop-Process
```

4.7 GNU/Linux commands equivalents

- which

Equivalent of *Nix 'which' command in Powershell?:

```
function which($name)
{
    Get-Command $name | Select-Object -ExpandProperty Definition
}
```

4.8 PowerShell Community Extensions (PSCX)

4.8.1 Symbolic Link

- New-Symlink
- New-Junction
- New-Hardlink

4.8.2 Path

- Get-PathVariable
- Set-PathVariable
- Add-PathVariable

4.9 Remoting

<http://superuser.com/questions/643120/windows-powershell-screen-equivalent/643606>

```
New-PSSession -ComputerName localhost
Get-PSSession -ComputerName localhost | Disconnect-PSSession
```

4.9.1 PowerShell Remoting vs ssh

- In PowerShell Remoting, basically everything happens in the local machine; you send cmdlets to remote machines and get the results back.
- With ssh, one logs into a remote machine, and then use a utility such as screen to make persistent sessions.
- With PowerShell Remoting, one generates PSSessions from the local machine, and then Enter those PSSessions via Enter-Pssession.

4.9.2 Enable Remoting

On the remote computer:

```
Enable-PSRemoting
```

4.9.3 Check the port on the remote machine

<http://stackoverflow.com/questions/5458565/powershell-remote-enabling>

On the local computer:

```
cd WSMAN:\localhost\Listener
WSMan:\localhost\Listener> dir
[cut]
WSMan:\localhost\Listener> cd .\Listener_1084132640
WSMan:\localhost\Listener\Listener_1084132640> dir
WSManConfig:
icrosoft.WSMan.Management\WSMan::localhost\Listener\Listener_1084132640
Name Value
----
Address *
Transport HTTP
Port 5985
```

4.9.4 Configure TrustedHosts

On the local computer:

```
Set-Item WSMAN:\localhost\Client\TrustedHosts *
Set-Item WSMAN:\localhost\Client\TrustedHosts $office
Restart-Service winrm
```

You can check it by:

```
Get-Item WSMAN:\localhost\Client\TrustedHosts
```

4.9.5 Connect to the remote machine

On the local computer:

```
$targetServer = "xxx.xx.xxx.xx"
$remotePowerShellPort = 5985

$connectionURI = ("http://{0}:{1}" -f $targetServer, $remotePowerShellPort)
```

Invoke-Command:

```
$remoteSession = New-PSSession -ConnectionURI $connectionURI

Invoke-Command -Session $remoteSession -ScriptBlock {Get-Process} -AsJob
```

Enter-PSSession:

```
Enter-PSSession -ConnectionURI $connectionURI

Exit-PSSession
```

Note: Enter-PSSession without session argument makes a temporary PSSession automatically, which is not persistent. You have to make a PSSession first via New-PSSession and then enter into that PSSession.

Note: In default WinRM configuration, you can skip `-ConnectionURI` and `http://` and use the ip address as the computer name.

With credentials

```
$Cred = Get-Credential "remotecomputername\username"  
Enter-PSSession XXX.XX.XX.XX -Credential $cred
```

4.9.6 Interactive Remote Session

Open a persistent interactive session:

```
$s = New-PSSession -ConnectionURI $ConnectionURI  
Enter-PSSession -Session $s
```

Disconnect from the session:

```
Exit-PSSession  
Disconnect-PSSession -Session $s -OutputBufferingMode Drop -IdleTimeoutSec 2147483647
```

Recover the session:

```
$session = Get-PSSession -ConnectionURI $ConnectionURI
```

You can re-enter the remoting session:

```
Enter-PSSession -Session $s
```

Note: It seems this session is limited in a sense that console outputs will not be shown correctly; for example, invoking `vim` will make console to hang. I haven't figured out how to fix this. It seems it is a inherent problem of remoting.

4.9.7 Remote Disconnected Sessions

about Remote Disconnected Sessions

1. Creates a session to the `ConnectionURI` computer:

```
> New-PSSession -ConnectionURI $ConnectionURI
```

```
> New-PSSession -ConnectionURI $ConnectionURI -Name "IPython Notebook Server"
```

2. To get the session, use the `ConnectionURI` parameter of `Get-PSSession` with a value of `$ConnectionURI`:

```
> Get-PSSession -ConnectionURI $ConnectionURI
```

3. To disconnect a `PSSession` use the `Disconnect-PSSession` cmdlet:

```
> Get-PSSession -ConnectionURI $ConnectionURI | Disconnect-PSSession
```

4. To connect a disconnected PSSession, use the Connect-PSSession cmdlet:

```
> Connect-PSSession -ConnectionURI $ConnectionURI -Name Session2
```

5. Run a command remotely in a disconnected session:

```
> Invoke-Command -ConnectionURI $ConnectionURI -InDisconnectedSession -  
↳ScriptBlock {ipython notebook --profile=nbserver}
```

6. Run a command remotely as a job in a disconnected session:

```
> $s = Invoke-Command -ConnectionURI $ConnectionURI -InDisconnectedSession -  
↳ScriptBlock {Start-Job -ScriptBlock {ipython notebook --profile=nbserver}}
```

- In this case, you need to do Receive-PSSession to be able to access the session:

```
> Receive-PSSession $s
```

4.9.8 Start A Remote Job that Returns the Results to the Local Computer (Asjob)

<http://technet.microsoft.com/en-us/library/hh849719.aspx>

Runs the command as a background job on a remote computer. Use this parameter to run commands that take an extensive time to complete.

When you use AsJob, the command returns an object that represents the job, and then displays the command prompt. You can continue to work in the session while the job completes. To manage the job, use the Job cmdlets. To get the job results, use the Receive-Job cmdlet.

For example:

```
$ Invoke-Command -ConnectionURI $ConnectionURI -ScriptBlock {python_  
↳C:\Users\joon\Dropbox\playground.py} -AsJob  
$ Invoke-Command -ConnectionURI $ConnectionURI -ScriptBlock {ipython notebook --  
↳profile=nbserver} -AsJob
```

Note: Problem with this is that since the results are returned to the local computer, once the local session is ended, the results are lost. I think the job gets lost as well.

4.9.9 Start A Remote Job that Keeps the Results on the Remote Computer

<http://technet.microsoft.com/en-us/library/hh847805.aspx>

I think this can be very useful.

1. Generate a new PSSession, or connect to an existing one. Let \$s denote the session variable.
2. Invoke a command as a Start-Job in the session:

```
Invoke-Command -Session $s -ScriptBlock {Start-Job -ScriptBlock {python_  
↳C:\Users\joon\Dropbox\playground.py}}
```

3. You can `Disconnect-PSSession -Session $s` freely, without killing the job.
4. To get list of jobs in the session `$s`:

```
Invoke-Command -Session $s -ScriptBlock {Get-Job}
```

5. To get the output from the job:

```
Invoke-Command -Session $s -ScriptBlock {Receive-Job JobId -keep}
```

4.9.10 Example: one session, multiple remote jobs

```
# Open a new PSSession
> $s = New-PSSession -ConnectionURI $ConnectionURI -Name Persistent

# Invoke command on the remote server
> {Start-Job -Name NBServer -ScriptBlock {ipython notebook --profile=nbserver}} | % {
↪ Invoke-Command -Session $s -ScriptBlock $_; };

# Disconnect $s with maximum IdleTimeoutSec
> Disconnect-PSSession -Session $s -OutputBufferingMode Drop -IdleTimeoutSec
↪ 2147483647

# Connct to $s
> Connect-PSSession $s

# Run additional jobs
> {Start-Job -Name Playground -ScriptBlock {python C:\Users\joon\Dropbox\playground.
↪ py}} | % { Invoke-Command -Session $s -ScriptBlock $_; };

# Get list of jobs in $s
> {Get-Job} | % { Invoke-Command -Session $s -ScriptBlock $_; };

# Get the output from the job with name Playground
> {Receive-Job -Name Playground -Keep} | % { Invoke-Command -Session $s -ScriptBlock
↪ $_; };
```

4.9.11 References

[An Introduction to PowerShell Remoting](#)

- [The Scipy Stack specification](#)

5.1 Resources

- [Enthought Canopy](#)
- [Anaconda](#)
- [Python\(x,y\)](#)
- [Unofficial Windows Binaries for Python Extension Packages](#)
 - For individual packages.
 - Also, it has both 32 and 64-bit packages.

All of the distributions come with MinGW.

If you need `openmp` support in Cython, then you should install [TDM-GCC](#)

Anaconda Installation

6.1 Download

Anaconda download page: <https://store.continuum.io/cshop/anaconda>

6.1.1 Path setup

```
$ export anaconda=C:\Anaconda
$ PYTHON=$anaconda
```

6.1.2 pylauncher

`pylauncher` for command line python script launching

6.2 Mayavi

Deprecated since version 1.7.0: Use `conda install mayavi` instead.

<http://www.lfd.uci.edu/~gohlke/pythonlibs/> maintains fantastic repository of binary Python libraries. To install Mayavi, you need to install the following two packages:

```
VTK-5.10.1.win-amd64-py2.7
ets-4.3.0.win-amd64-py2.7
```

- ets ets
- VTK

For some reason, if you install `VTK-Qt-5.10.1.win-amd64-py2.7`, it yields DLL error.

And also, install `configobj` if you get import failure with `configobj`:

```
pip install configobj
```

6.3 ViTables

From <http://www.lfd.uci.edu/~gohlke/pythonlibs/>, download the following packages, `PyQt` and `ViTables`:

```
PyQt-Py2.7-x64-gpl-4.9.6-1.exe  
ViTables-2.1.win-amd64-py2.7.exe
```

You can run `ViTables` with the following commands (use it as shortcut target):

```
C:\Anaconda\pythonw.exe "C:\Anaconda\Scripts\vitables"
```

6.4 Update Packages

6.4.1 Using conda

Accelerate

<http://docs.continuum.io/accelerate/>

```
$ conda update conda  
$ conda install accelerate
```

Others

Note: Make sure you run `conda` with all python app closed. Otherwise, it might fail with permission error.

```
$ conda remove pip  
$ conda update distribute cython matplotlib pandas scipy  
$ conda install pip
```

6.4.2 Using pip

```
$ pip install bottleneck virtualenvwrapper-powershell
```


6.4.3 Manual

CythonGSL

```
$ git clone git@github.com:joonro/CythonGSL.git
$ cd CythonGSL
$ git remote add upstream git://github.com/twiecki/CythonGSL.git
$ python setup.py install
```

IPython

```
$ rm -rf $anaconda/Lib/site-packages/IPython $anaconda/Lib/site-packages/ipython*
$ python setup.py install
```

Statsmodels

```
$ rm -rf $anaconda/Lib/site-packages/statsmodels*
$ pip install patsy

$ git clone git@github.com:joonro/statsmodels.git
$ cd Statsmodels
$ python setup.py install
```

Cython

```
$ rm -rf $anaconda/Lib/site-packages/Cython* $anaconda/Lib/site-packages/cython.*
$ python setup.py install
```

6.4.4 Virtualenv

virtualenvwrapper-powershell is installed as a module in `~/Documents/WindowsPowerShell/Modules`. To load the module, do `import-module virtualenvwrapper`.

Also, it will complain about not finding `Python InstallPath` from the registry. Copy `InstallPath` and `PythonPath` from `HKLM\Software\Python\PythonCore\2.7` to `HKCU\Software\Python\PythonCore\2.7`.

7.1 General

7.2 Git

- Git for Windows
- Install it with default settings (I tend to use option 3 so unix commands are available in other shell)
- Git Bash command line:

```
C:\Windows\SysWOW64\cmd.exe /c "C:\Program Files (x86)\Git\bin\sh.exe" --login -i  
↵
```

- Just use Chocolatey to install it:
choco install git
- When you try to update it with Chocolatey, you might get an error saying Setup cannot continue until you close at least those applications in the list that are marked as "closing is required". Stop ssh.exe and try it again:

```
Get-Process ssh.exe | Stop-Process
```

7.3 PowerShell

7.3.1 posh-git

<https://github.com/dahlbyk/posh-git>

A set of PowerShell scripts which provide Git/PowerShell integration

7.4 Terminal

- PowerShell
- ConsoleZ
- ConEmu

7.5 Libraries

7.5.1 GSL

Download

You can download binary GSL from [oscats](#) projects. Make sure you get the correct architecture. (Usually 64bit)

Environmental Variables

Add the path to bin directory of your GSL installation to your PATH Environmental variable:

```
setx -m PATH "$env:path;C:\lib64\gsl\bin"
```

Add an environmental variable LIB_GSL with the path to your GSL installation:

```
setx -m LIB_GSL "C:/lib64/gsl"
```

Note: You need to run above commands in administrator mode in CMD.

7.6 Compilers

7.6.1 Windows SDK C/C++

<https://github.com/cython/cython/wiki/64BitCythonExtensionsOnWindows>
[python-64-bit-on-windows-part-2-building-packages](https://github.com/cython/cython/wiki/python-64-bit-on-windows-part-2-building-packages)

<http://rexdouglass.com/>

7.6.2 MinGW with OpenMP support

MinGW doesn't come with openmp by default so if it's not included in the Anaconda version you'll probably have to find the correct installer from the MinGW site and install it.

<https://groups.google.com/a/continuum.io/forum/#!topic/anaconda/c-4u6C29hYM>

Warning: The cython [wiki](#) Does not recommend using MinGW for the 64bit compiler.

Install TDM-GCC

TDM-GCC is a compiler suite for Windows.

It combines the most recent stable release of the GCC toolset with the free and open-source MinGW or MinGW-w64 runtime APIs to create a LIBRE alternative to Microsoft's compiler and platform SDK

openmp support must be manually added during installation. Expand Components - gcc and check openmp:

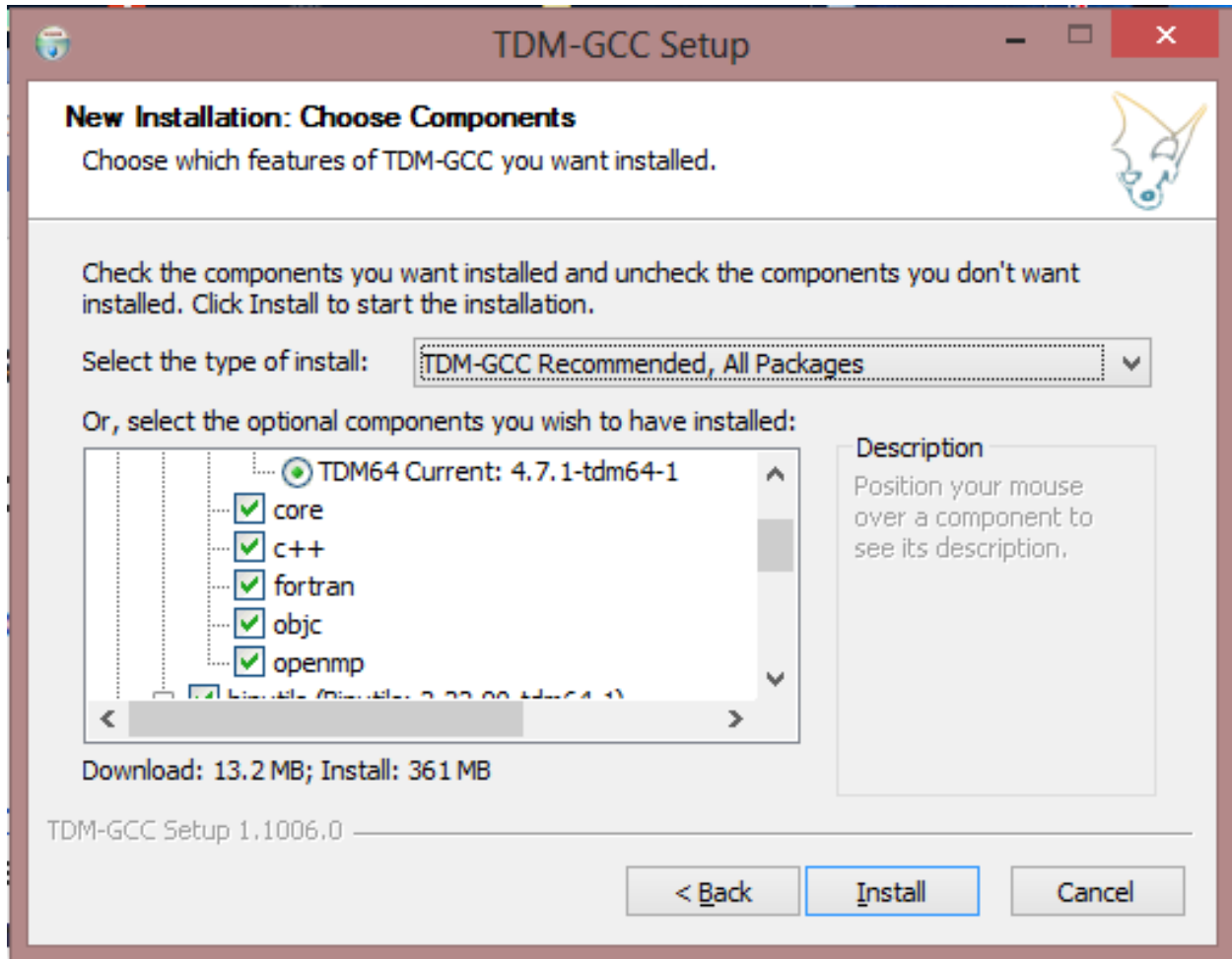


Fig. 7.1: TDM-GCC Installation

8.1 Global location of .bib file.

<http://tex.stackexchange.com/questions/49126/common-bib-file-for-several-collaborators>

As in GNU/Linux, you can make `texmf\bibtex\bib` directory under the user's profile directory (`C:\Users\username`).

```
mkdir ~/texmf
mkdir ~/texmf/bibtex
mkdir ~/texmf/bibtex/bib
```

The following is copied from the stackexchange entry:

1. On Windows with TeX Live, make a `texmf\bibtex\bib` directory under the user's profile directory (`C:\Users\username` by default on Windows Vista and higher).
2. Place the .bib file in that folder.
3. Place any custom bibliography styles in a `texmf\bibtex\bst` directory under the user's profile directory.
4. Pull up a command prompt, and run `mktexlsr texmf` from the profile directory (I don't remember if TeX Live automatically has this in the path or not, so you may need to run it as `C:\texlive\2011\bin\win32\mktexlsr texmf` or similar instead).
5. Write your documents wherever, and use the regular bibliography and related commands as usual. No need for paths, since your personal `texmf` tree will automatically be searched for support files.

8.2 Inverse search with LyX

<http://wiki.lyx.org/LyX/SyncTeX#toc3>

1. Set Tools > Preferences > Paths > LyxServer pipe to `\\.\pipe\lyxpipe`.
2. In Document > Settings > Output, check Synchronize with output.

3. Create a batch-file named `lyxeditor.cmd` and save it to one of the locations in your `PATH` Windows environmental variable:

```
@echo off
SETLOCAL enabledelayedexpansion
set file=%1
set row=%2
REM remove quotes from variables
set file=!file:!="!
set row=!row:!="!
%comspec% /q /c (@echo LYXCMD:revdvi:server-goto-file-row:%file% %row%)> \\.
↪\pipe\lyxpipe.in&&type \\. \pipe\lyxpipe.out
endlocal
```

8.2.1 AutoHotkey Script

The above cmd works well, but it shows annoying black cmd window everytime when you invoke the script. With a simple AutoHotkey script, one can not only suppress this window, but also activate LyX windows after the inverse search.

Create a AutoHotkey script named `lyx-inverse-search.ahk` with the following code and save it to the same location to `lyxeditor.cmd` and compile it with AutoHotkey to generate `lyx-inverse-search.exe`:

```
SetTitleMatchMode, RegEx
Run, lyxeditor.cmd "%1%" "%2%",, Hide
WinActivate, ^LyX,,,
```

If you don't have AutoHotkey installed, you can download the compiled `exe`.

8.2.2 SumatraPDF

<http://wiki.lyx.org/Windows/LyXWinTips#toc6>

1. Download and install Sumatra PDF.
2. In LyX go to Tools > Preferences > Paths and add the install location to `PATH` prefix. Most likely this is `C:\Program Files\SumatraPDF`.

Note: If you use Chocolatey to install Sumatr PDF, the `PATH` is `C:\Chocolatey\bin`

3. In Tools > Preferences > File Handling > File Formats select PDF (pdflatex) from the list of formats and modify Viewer to the following:

```
SumatraPDF -reuse-instance -inverse-search "lyx-inverse-search.exe \"%f\" \"%1\""
```

4. If you don't have the compiled AutoHotkey script from above, use the following:

```
SumatraPDF -reuse-instance -inverse-search "lyxeditor.cmd \"%f\" \"%1\""
```

8.2.3 Okular

1. In LyX go to Tools > Preferences > Paths and add the location of `okular.exe` to `PATH` prefix. Most likely this is `C:\Program Files (x86)\KDE\bin`

2. In Tools > Preferences > File Handling > File Formats select PDF (pdflatex) from the list of formats and modify Viewer to:

```
okular --unique
```

3. In Okular, Settings > Configure Okular > Editor, choose Custom Text Editor and input the following:

```
lyx-inverse-search.exe "%f" "%l"
```

If you don't have the compiled AutoHotkey script from above, use the following:

```
lyxeditor.cmd "%f" "%l"
```

Note: If `lyxeditor.cmd` or `lyx-inverse-search.exe` is not in your system PATH, then you have to specify the full path here.

8.3 Forward search with LyX

8.3.1 SumatraPDF

Forward search setting is easier. Make sure you have the *PATH* to Sumatra PDF in Tools > Preferences > Paths. Then in Tools > Preferences > Output > General, choose SumatraPDF `-reuse-instance $$o -forward-search $$t $$n`, which should be built-in.

Note: If you have master and child document structure, the forward search only works when you compile the full document. If you only compile current child document forward search will not work.

8.3.2 Okular

```
okular --unique "file:$$o#src:$$n $$f"
```

8.4 Mendeley Settings

Tools -> Options -> BibTeX, check Enable BibTeX syncing. Create one BibTeX file per collection. Path, put `C:\Users\joon\txmf\bibtex\bib`.

9.1 LyX settings

In `C:\Users\User\AppData\Roaming\LyX2.X folder:

```
preferences # general preferences  
pwl_english.dict # personal word list  
bind\user.bind # key binding
```

9.2 Outlook

*.ost files in In `C:\Users\User\AppData\Local\Microsoft\Outlook folder.

9.3 Windows 8 Apps

Manga Z

```
C:\Users\joon\AppData\Local\Packages\36032Twincubesstudio.MangaZ_3znphxp8kcdy
```


10.1 General

I suggest that you run SFC scan on the computer to see if fixes the issue. SFC (System File Checker) checks for any corrupt system files and tries to repair them. To do so, follow these steps.

1. At the Start screen, type `cmd`.
2. You will find Command Prompt, right click on it and select `Run as administrator` from the bottom of the screen. At the command prompt, type the following command, and then press enter: `sfc /scannow`

The `sfc /scannow` command scans all protected system files and replaces incorrect versions with correct Microsoft versions.

- Error log is located at:

```
C:\$Windows.~BT\Sources\Panther\setuperr.log
```

10.2 Skydrive

10.2.1 Disable Skydrive in Windows 8.1

<http://www.ghacks.net/2013/11/02/disable-skydrive-windows-8-1/>:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\Skydrive
```

You may need to create the Skydrive folder here. If you do, right-click on Windows and select `New > Key` and name it `Skydrive`.

Right-click on `SkyDrive` and select `New > Dword (32-bit value)` and name it `DisableFileSync`.

Double-click the new parameter and change its value to `1`.

10.3 Network

- Resolving proxy
- How to Delete or Forget Wireless Network Profiles in Windows 8.1:

```
netsh wlan show profiles
netsh wlan delete profile name="profile name"
```

10.4 0xd0000225 switch to live account

This user Wasn't Added to this PC code

Add pku2u and livessp to the Security Packages REG_MULTI_SZ key under regkey HKLM\System\CurrentControlSet\Control\Lsa, and reboot.

Windows 8 sleeps after few minutes when awoken by external USB keyboard

Forum

1. Get registry from [Here](#) and add "System unattended sleep timeout" to Advanced Power Settings
2. Following [Here](#), increase system unattended sleep timeout.

CHAPTER 11

Indices and tables

- `genindex`
- `modindex`
- `search`